STPC
Video Input Port Writer's Guides

Issue 1.0

September 1999

Information provided is believed to be accurate and reliable. However, ST Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringements of patents or other rights of third parties which may result from its use. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied.

STMicroelectronics
Technoparc du Pays de Gex - B.P. 112
165, rue Edouard Branly
01630 Saint Genis Pouilly (France)

1. INTRODUCTION

STPC Video Input Port can capture a video stream and store images inside the Frame Buffer area of Unified Memory Architecture.

Video Input Port supports a large range of resolutions and refresh rates and two types of synchronization signals. Vertical and horizontal synchronization can be encapsulated inside the video stream as specified in the ITU-R-656 standard (EAV and SAV code) or driven by electrical signals. When the synchronization is driven by an electrical signal, the vertical synchronization is done using bottom/top method: the vertical signal switch to indicate a frame change (odd/even in interlaced stream or bottom/top) and not a blanking signal.

Warning: the STPC Video Input Port supports only a subset of ITU-R-656 where horizontal and vertical blanking signals are constant. The size of the blanking signals are fixed by programming registers.

2. ACCESSING VIDEO INPUT PORT REGISTERS

Before accessing the Video Input Port registers, the extended graphic functionalities have to be enabled in the CRTC Register 0x1Fh (bit 7 set to 1).

After setting this bit to 1, the Video Input Port registers are mapped in address: ((GBASE | 8) << 24) + 0x600000h where GBASE is the value of the CRTC register 0x20h.

The CRTC registers 0x1Fh and 0x20h are CRTC Extended registers and can be accessed only after enabling extended registers by writing 0x57h to the Sequencer register 0x6h. The CRTC registers are accessible by indexed I/O in 0x3D4h/0x3D5h and the Sequencer registers using indexed I/O in 0x3C4h/0x3C5h.

2.1. INITIALIZING VIDEO INPUT PORT

Four VIP registers are used to configure the video source and have to be adapted to your own source:

Horizontal timing register (offset 0x30h),

Vertical timing register (offset 0x34h) and

Timing generator register 1 and 2 (offset 0x28h and 0x2Ch).

2.2. CONFIGURING VIP IN ITU-R-601 MODE (ELECTRICAL SIGNAL FOR SYNCHRONIZATION)

In ITU-R-601 mode, VIP uses external timing signals from the VIP connector (HSYNC and B/T) to synchronize the video. The HSYNC signal is active when the data is active video and reset when the data is blank. The B/T signal switch is used to differentiate odd frames from even frames in interlaced mode or to differentiate bottom and top of the frame in none interlaced mode.

Video timing has to be enabled in the Timing Generator register 1 (bit 31 to 1), HSYNC and B/T have to be set as inputs (bit 28 and 29 set to 0), the Genlock mode to B/T and HSYNC signal (value 001 in bits 21-23). B/T and HSYNC can be active on low or high level and bit 25 and 26 have to be adapted to your B/T and HSYNC signals. Other parameters of the Timing Generator registers 1 and 2 are available to adjust the synchronization signal edges (to detect a new frame) and have to be modified if odd and even frames are not correctly detected (random inversion between odd and even field).

Horizontal and Vertical Timing registers have to be adjusted to suppress unused pixels on the borders of the video. Each of these registers contain a start offset, a stop offset and the total number of pixels in a line. The stop offset is valid only if the total number of pixels is not reached before the stop offset. The timings are specific to a video source and have to be carefully adjusted by playing with these two registers.

Bits 16 to 18 of the Configuration register (0x4h) have to be set to ITU-R-601 mode (011).

2.3. CONFIGURING VIP IN ITU-R-656 MODE

In ITU-R-656 mode, the VIP uses values encoded in the data stream to determine horizontal and vertical synchronization. A SAV (Start of Active Video) data indicates the start of a line and an EAV (End of Active Video) data indicates the end of the line.

To use the ITU-R-656 mode, the genlock mode has to be set to 010 in bits 21 to 23 of the Timing Generator register 1 and HSYNC and B/T signal configuration are not used anymore. It is possible to configure these two signals in the output (bits 28 and 29 of the Timing Generator register 1) to generate HSYNC and B/T signal from SAV and EAV code for external use (synchronize a second video source for example). In this case, HSYNC and B/T signal parameters have to be set up in Timing Generator registers 1 and 2 as describe in ITU-R-601 mode.

Vertical and Horizontal Timing registers are used to suppress unused pixels on the borders of the video image as describe in ITU-R-601 section. The vertical stop and total parameters are not used anymore and all lines of the video are automatically written to the memory.

Bits 16 to 18 of the Configuration register (0x4h) have to be set to ITU-R-656 mode (000).

2.4. CONFIGURATION IN BOTH ITU-R-601 & ITU-R-656

For both ITU-R-601 and ITU-R-656 mode, data can be written directly to memory or swapped to be compatible with YUV422 pixel format of the STPC Video Pipeline (video overlay). Data values from the input stream are UYUV data and video data supported by Video Pipeline are YUYV format. ITU-R-601/656 Swap is enabled by setting 0 to bit 5 of configuration register (offset 0x4h).

3. ORGANIZING VIDEO DATA IN MEMORY

In order to support interlaced video, the video frame is separated into two fields, managed by a different set of registers. The field 0 correspond to the odd frame in an interlaced video stream and the field 1 correspond to the even frame in the interlaced video stream. In none interlaced stream, field 0 and field 1 correspond to top and bottom part of a frame (bottom/top method) or to two consecutive frames.

Field 0 and field 1 are configured by two base addresses (registers 0x0Ch for field 0 and 0x10h for field 1) which corresponds to the offset in the Frame Buffer and a common pitch register (register 0x14h) which indicate the number of bytes between two lines of a video frame. Video can't be written in the system memory but only in Frame Buffer area (area of 128Kb to 4Mb of memory reserved for video and graphics data).

3.1. CAPTURE VIDEO IN NONE INTERLACED MODE

In none interlaced mode, field 0 and field 1 base addresses have to be put to the same value. Pitch registers have to be set to at least twice the video width (1 pixel is 2 bytes) to be sure the next line will not erase data from previous one.

3.2. CAPTURE VIDEO IN BOTTOM/TOP MODE

Some video sources use field 0 for the top of the frame and field 1 for the bottom. In this case the base address of field 1 has to be set to the base address of field 0 plus one half of the frame size (pitch * number of line / 2).

3.3. CAPTURE A FULL INTERLACED VIDEO

Data in memory has to be set following this pattern:

Table 3-1. Memory Data pattern

W

H

0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0

Where 0 is field 0 and 1 field 1.

The base address of field 1 has to be set to base address of field 0 plus the size, in bytes, of a video line. The Pitch register has to be set to twice the size of a line to have a hole of 1 line for the other field.

3.4. CAPTURE ONLY ONE FIELD OF AN INTERLACED STREAM

Bits 9 to 7 of the Configuration register (0x4h) select the field to be written in memory (001: only field 0, 010: only field 1, 011: both field 0 and field 1). If only one field is captured, the number of captured lines is divided by two and the scale between width and height of the image is not kept. To avoid this problem, every other pixel can be suppressed enabling the horizontal decimator (bit 20 of Configuration register set to 1) in order to keep a good ratio between width and height.

Capturing only one field decreases the resolution of video but can increase the quality when displaying video on a monitor. When using standard interlaced mode on a monitor, both interlaced images are displayed on the screen virtually simultaneously causing a the fractioning of the image and a separation of the individual scan lines. This results in a substantial reduction of the image quality.

3.5. STARTING THE VIDEO CAPTURE

Video capture can be started in different ways by playing with the Configuration register (register 0x4h):

Bit 21 can be set to 1 to invert field 0 and field 1 (odd and even frame).

Bits 10 and 11 can be used to decrease the video capture rate and write in memory only one frame in two, one frame in three or one frame in four.

To use an external clock as a pixel clock (this is the normal use), bits 22 and 23 of the Configuration register have to be set to 0 & 1. It is safer to configure all parameters of the Configuration register before switching to external clock mode. If no external clock is detected, the video capture will not start. This can be checked reading bit 9 of the Status register.

To start video capture, bit 0 and 4 have to be set to 1. To end video capture after the current frame, bit 4 have to be set to 0. Bit 7 of Status register (0x8h) will be cleared by the end of frame. To stop video urgently, bit 0 can be set to 0.

When video is started, the video frame is refreshed in memory continuously, as long as the Video Input Port is not stopped.

4. SYNCHRONIZATION BY POLLING

Different bits of Status register (0x8h) can be used to synchronize software with the video frame:

bit 8 value is 0 when VIP is processing active video and 1 when VIP is processing vertical blank.

bit 4 indicate the field in progress.

These 2 bits can be used to wait for the end of a frame.

5. SYNCHRONIZATION BY INTERRUPT

STPC VIP can manage 3 types of interrupts for synchronization,

End of field interrupt (odd or even frame in interlaced or end of frame in none interlaced)

End of vertical blank signal interrupt of.

Buffer full interrupt (finish to fill the field 0 or field 1 buffer)

and 1 overflow interrupt to system device when the VIP is not able to process all the incoming data.

This 4th interrupt can be enabled by writing 1 in one of the bits 31 to 28 bits of the Configuration register or disabled by writing 1 in one of the 37 to 24 bits. All these interrupts use the same interrupt line defined in the chipset register (0x22h/0x23h) number 0x58h. This interrupt is shared by the VIP and the CRTC. After receiving one of the interrupts, the ISR has to check bit 0, 1, 2, 5 and 6 of Status register to determine the interrupt source. After that, the corresponding bit of the Status register has to be cleared to acknowledge the interrupt. The next interrupt of same type will not be handle before the first one is cleared.

6. DOUBLE-BUFFERING METHOD

Double-buffering is used to enhance the display quality when video stream capture is not synchronized with the display. When the STPC Video Pipeline is used to display the video, it does so at the refresh rate of the screen, when capturing for the memory, the input stream refresh rate is used.

The double buffering method is used to assure the current video frame is not updated by the Video Input Port in the same time as Video Pipeline (or CPU or any other hardware) takes the data from memory. Video Input Port captures a frame in two different memory buffers (this means two different location inside the Frame Buffer). The buffer filled by the Video Input Port is hidden during the capture phase and, at the end of the capture of a frame (detected by interrupt or by polling), the Video Pipeline base address is switched to the hidden buffer to display it (or CPU can start to process the data).

7. PROBLEM OF LOST OF SYNCHRONIZATION

In ITU-R-656 mode, STPC Video Input Port is designed to process a clean video stream and in particular concerning EAV (End of Active Video) code. The Video Input Port stop capturing a frame to start the next frame only when receiving appropriate EAV signal. If no EAV signal is received, Video Input Port will continue to write data inside the Frame Buffer and erase the other video and graphic data stored inside the Frame Buffer and even a part of system memory when the size of the Frame Buffer is less than 4Mb.

This problem often appears when source video is externally switch on, switch off, connected or disconnected when Video Input Port is active. This could happen when the quality of video input stream is poor.

There are different way to fix this problem:

- by switching off the Video Input Port before doing modification on the video input stream.
- by ensuring the external hardware will never miss an EAV when the video clock is active.
- by detecting the Video Input Port overrun in the Frame Buffer. Regularly read a memory value placed after the video frame in the Frame Buffer and detect when this value is modified by the memory overrun. The Video Input Port can be stopped when an overrun is detected. This can be done in a background task dedicated to this task or in any other routine regularly called by the system (a timer interruption routine for example).

- by detecting a video frame longer than a normal frame. This can be done by using end of frame interrupt. A time out interrupt is initialized at the end of every frame. The time out duration is a bit higher than a frame duration, and if the time out interrupt arrives before the next end of frame interrupt, the Video Input Port is stopped.

The 2 last method requires free space to be reserved in the Frame Buffer after the video frame to avoid problems of late detection of overrun. The size of this reserved space should be adjusted to the maximum delay of overrun detection.

8. HOW TO USE VIDEO INPUT PORT TO

8.1. CAPTURE OR MODIFY A VIDEO FRAME

Video data is available in the Frame Buffer memory and can be modified or transferred into system memory or a file. The physical address of the video data in memory is ((GBASE | 8) << 24) + 0x800000h. The way to get GBASE value is the same as the one to access Video Input Port register.

The format of video inside the memory is the same as the digital input stream, eventually swapped two bytes by two bytes if the byte swap option is enabled.

Usually this format is YUV422.

2 pixels are described by 32 bits in the following way.

byte 1: Y1 value (luminance)

byte 2: U value (chrominance)

byte 3: Y2 value (luminance)

byte 4: V value (chrominance)

First pixels use Y1UV

Second pixels use Y2UV

This means two consecutive pixels use the same value of chrominance and only the luminances change.

When pixel values are read from memory in 32 bit (the most efficient method) to be modified, the software needs to take care of the fact the Intel CPU swaps bytes when it reads a value from memory (little indian method).

To be sure the video data is not updated by the Video Input Port at the same time it is processed or transferred by the CPU, the software has to use the double buffering method.

8.2. DO REAL-TIME PROCESSING ON VIDEO

Video can be processed and displayed in real time using two buffers in the Frame Buffer. The Video Input Port captures a video frame in the first buffer, the CPU read data from this buffer, process it and writes it into the second buffer and the Video Pipeline displays the second buffer. To synchronize the whole stream of data and ensuring the video is processed only when a full frame is ready, the double buffering method can be used both for the capture phase (the CPU starts to process a frame when the full frame is captured) and the displaying phase (Video Pipeline displays a frame only when fully processed by the CPU).

The format of video data inside memory is usually YUV422 as described above.

CPU Read and write accesses are often a bottleneck for real-time performance issues. To get the best performance, CPU should do only 32 bit accesses to Frame Buffer and this access should be a 32 bits aligned access. The STPC Graphics Engine host-to-screen feature can be used to accelerate write accesses inside the Frame Buffer. This method is describe in Graphics Engine Writers Guide.

Note; CPU access to the Frame Buffer memory is not catchable and slower than access to system memory.

9. GLOSSARY

Frame Buffer :

The Frame Buffer is an area of memory reserved for graphics and video. This area can be set between 128Kb to 4Mb of memory and the Graphics Engine can work only in this area. Frame Buffer memory is equivalent to a video memory in a none UMA architecture.

Host memory :

Host memory correspond to the system memory, visible by the operationg system.

Data Port :

The Data Port FIFO is used to send data to the Graphics Engine when this data is not directly accessible. The Data Port FIFO is mapped in memory and is accessible using a physical memory address in the Graphics Engine area.

Pixel depth :

The pixel depth is the number of Bytes for one pixel of a graphics area. The pixel depth defines the number of colors managed in this area.

Unified Memory architecture :

Unified Memory Architecture (UMA) is a memory architecture where system memory, graphics memory and video memory is physically grouped inside same memory chip, using the same memory bus.

CRTC :

The Cathodic Ray Tube Controller get data from the Frame Buffer to display them on a monitor or a TV.

Video Pipeline :

Video overlay used to display video on screen.

Video Input Port :

Video input used to capture a video stream inside the Frame Buffer.

Bitblit :

Operation of image (rectangle area) copied inside memory.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

© 1999 STMicroelectronics - All Rights Reserved

The ST logo is a registered trademark of STMicroelectronics.

All other names are the property of their respective owners.

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - China - France - Germany - Italy - Japan - Korea - Malaysia - Malta - Mexico - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.